Exhibit 2

U.S. Patent No. 8,549,339 ("'339 Patent")

Accused Products

Qualcomm's products comprising one or more SoC each comprising two or more sets of processors implementing or based on the DynamIQ Shared Unit architecture (*e.g.*, ARMv8.2, ARMv9 ARMv9.2, and successors) or big.LITTLE architecture, including without limitation the Snapdragon 8 Gen 2 and the Snapdragon 835 Mobile Platform.

Claim 1

Claim 1	Accused Products
[1pre] 1. A multi-core processor, comprising:	To the extent the preamble is limiting, each Accused Product comprises a multi-core processor. For example, Snapdragon 8 Gen 2 Octa-Core processor contains eight cores implementing the ARM DynamIQ Shared Unit-110 architecture. In a further example, the Snapdragon 835 Mobile Platform contains eight cores implementing the ARM big.LITTLE architecture. *See, e.g.*:

Claim 1	Accused Products

	Snapdragon 8 Gen 2 (2023) [edit]
	The **Snapdragon 8 Gen 2** was announced on November 15, 2022.[252]
	Notable features over its predecessor (8 Gen 1):
	• 4nm (TSMC N4) process
	• Support up to 16 GB LPDDR5X 4200 MHz
	• Support UFS 4.0
	• CPU features
	  • 1 Kryo Prime (ARM Cortex-X3), up to 3.36 GHz. Prime core
	  • 1MB L2 cache
	  • Only 64-bit support [253]
	  • 2 Kryo Gold (ARM Cortex-A715), up to 2.8 GHz. High Performance cores
	  • Only 64-bit support [253]
	  • 2 Kryo Gold (ARM Cortex-A710), up to 2.8 GHz. Performance cores
	  • 32-bit and 64-bit support[253]
	  • 3 Kryo Silver (ARM Cortex-A510), up to 2 GHz. Efficiency cores
	  • 32-bit and 64-bit support[253]
	  • 35% performance uplift and 40% power efficiency improvement
	  • 8 MB system-level cache
	https://en.wikipedia.org/wiki/List_of_Qualcomm_Snapdragon_systems_on_chips#Snapdragon_8_Gen_2_(2023)

Claim 1	Accused Products

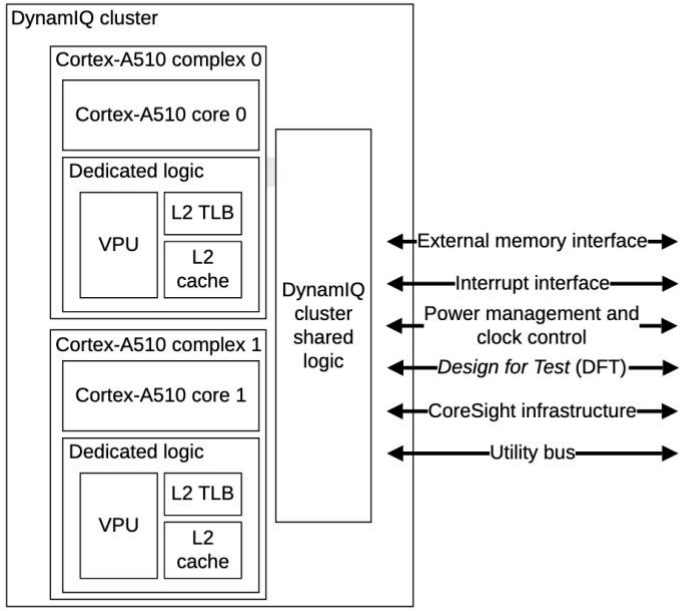
Model number	Product Name	Fab	Die size	CPU
MSM8998[246]	Snapdragon 835	10 nm FinFET (Samsung 10LPE)	72.3 mm^2	4 + 4 cores Kryo 280 (2.45 GHz Cortex-A73 + 1.9 GHz Cortex-A53)

https://en.wikipedia.org/wiki/List_of_Qualcomm_Snapdragon_systems_on_chips#Snapdragon_800_series_(2013%E2%80%932021)

Claim 1	Accused Products
	DynamIQ Shared Unit-110 First DynamIQ Shared Unit of the Armv9 generation, up to 12 cores supported, up to 16MB L3, and enhanced power management features. **Supporting:** • Cortex-X3 • Cortex-X2 • Cortex-A715 • Cortex-A710 • Cortex-A510 https://www.arm.com/technologies/dynamiq

Claim 1	Accused Products

Figure 2-2: Example configuration with two Cortex®-A510 single-core complexes

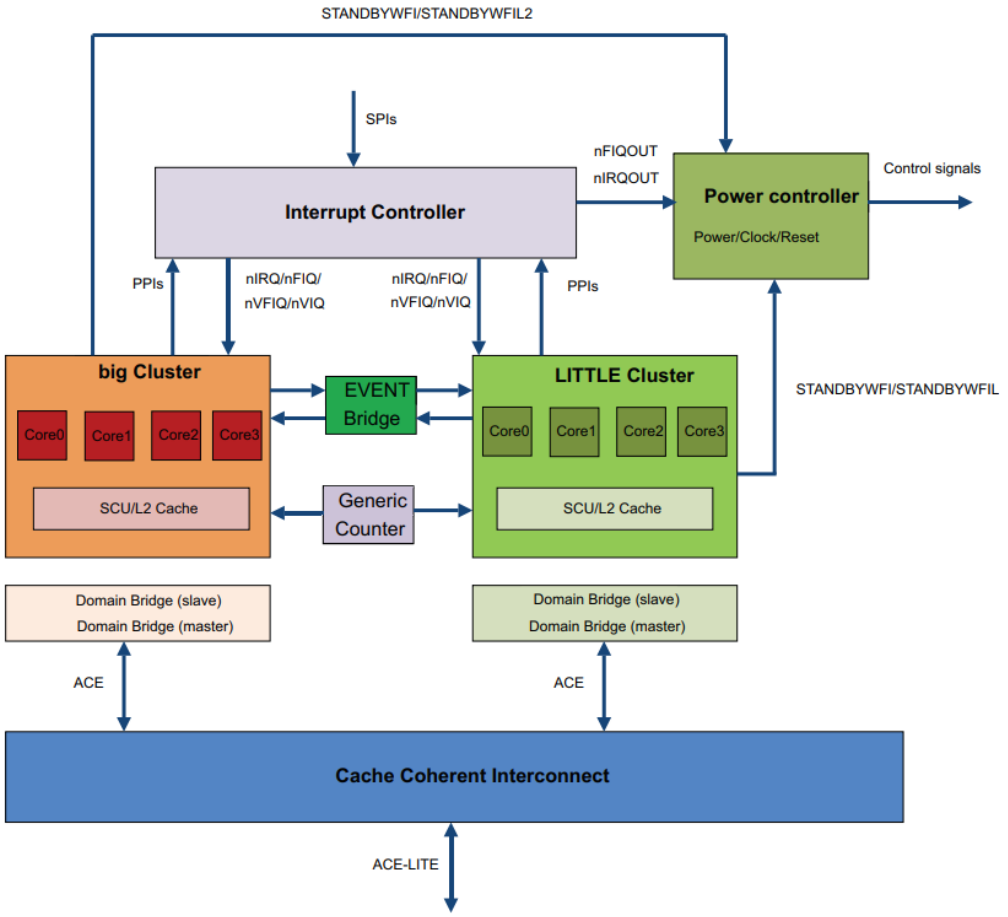


Arm Cortex-A510 Core Technical Reference Manual at p. 23

> Our initial testing shows that Snapdragon 835's Kryo 280 CPU is an octa-core, big.LITTLE configuration with four semi-custom A73 "performance" cores and four semi-custom A53 "efficiency" cores. Kryo 280's performance cores are pretty much equivalent to Kirin 960's A73 cores in both integer and floating-point IPC, but comparing them to Snapdragon 820's Kryo CPU shows mixed results: integer IPC improves but floating-point regresses.

https://www.anandtech.com/show/11201/qualcomm-snapdragon-835-performance-preview/6

Claim 1	Accused Products
	1.2 Document scope This application note focuses on the following processors and highlights important issues when powering up or powering down processor cores and clusters on an SoC. • Cortex®-A7. • Cortex®-A15. • Cortex®-A17. • Cortex®-A53. • Cortex®-A57. • Cortex®-A72. • Cortex®-A73. "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/ **2.1 ARM® big.LITTLE™ system example** A big.LITTLE system uses two different types of cores that are combined in a coherent system. big cores are designed for high performance while LITTLE cores are designed for high energy efficiency. The big cores are used for resource-intensive software threads, and energy-efficient LITTLE cores handle low-intensity software threads that use fewer compute resources. The result is high energy efficiency and high performance. "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
	 **Figure 2-1 A typical big.LITTLE system** "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
[1a] a first set of processor cores of the multi-core processor, wherein each processor core from the first set of processor cores is configured to dynamically receive a first supply voltage and a first output clock signal of a first phase lock loop (PLL) having a first clock signal as input;	Each Accused Product comprises a first set of processor cores of the multi-core processor, wherein each processor core from the first set of processor cores is configured to dynamically receive a first supply voltage and a first output clock signal of a first phase lock loop (PLL) having a first clock signal as input. For example, the Snapdragon 8 Gen 2 Octa-Core includes a first set of high efficiency processor cores (two or more ARM Cortex-A510 cores). The processor cores in the first set receive a dynamic supply voltage and a first output clock signal of a first PLL having a first clock signal as input. For another example, the Snapdragon 835 Mobile Platform includes a first set of high efficiency processor cores (two or more A53 cores). The processor core(s) in the first set receive a dynamic supply voltage and a first output clock signal of a first PLL having a first clock signal as input. ARM documentation for the big.LITTLE and DynamIQ architectures used in the Accused Products directly shows that each core cluster receives its own clock domain. At the time the Accused Products were designed, it was typical to produce this clock using a PLL that has a corresponding clock input. Furthermore, ARM documentation for an earlier, related device (the Cortex-A15_A7 MPCore test chip, which also has independent clock domains for different CPU clusters) shows each CPU cluster receiving an output clock signal from a PLL having a corresponding clock signal as input (*e.g.* from an oscillator). It is therefore substantially likely that each Accused Product specifically receives a first output clock signal of a first PLL having a first clock signal as input. *See, e.g.*:

Claim 1	Accused Products

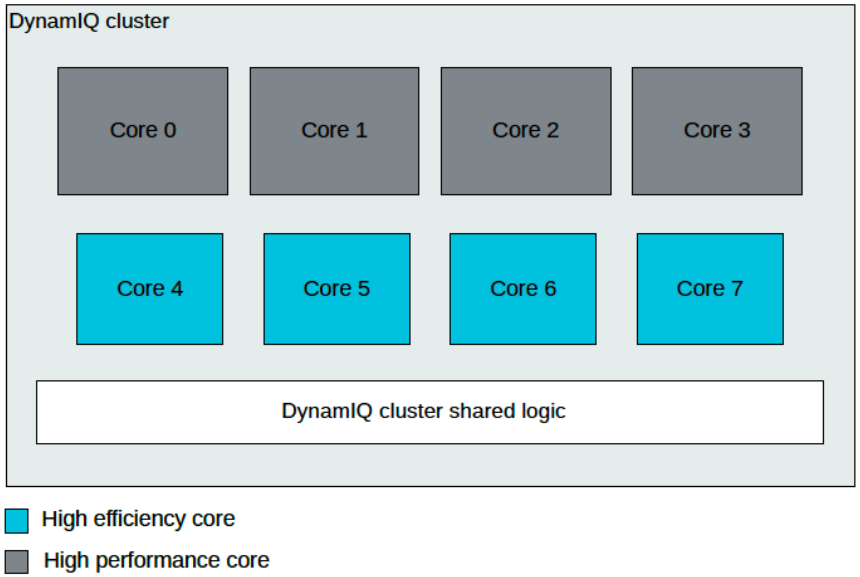
	Snapdragon 8 Gen 2 (2023) [edit]
	The **Snapdragon 8 Gen 2** was announced on November 15, 2022.[252]
	Notable features over its predecessor (8 Gen 1):
	• 4nm (TSMC N4) process
	• Support up to 16 GB LPDDR5X 4200 MHz
	• Support UFS 4.0
	• CPU features
	  • 1 Kryo Prime (ARM Cortex-X3), up to 3.36 GHz. Prime core
	  • 1MB L2 cache
	  • Only 64-bit support [253]
	  • 2 Kryo Gold (ARM Cortex-A715), up to 2.8 GHz. High Performance cores
	  • Only 64-bit support [253]
	  • 2 Kryo Gold (ARM Cortex-A710), up to 2.8 GHz. Performance cores
	  • 32-bit and 64-bit support[253]
	  • 3 Kryo Silver (ARM Cortex-A510), up to 2 GHz. Efficiency cores
	  • 32-bit and 64-bit support[253]
	  • 35% performance uplift and 40% power efficiency improvement
	  • 8 MB system-level cache
	https://en.wikipedia.org/wiki/List_of_Qualcomm_Snapdragon_systems_on_chips#Snapdragon_8_Gen_2_(2023)

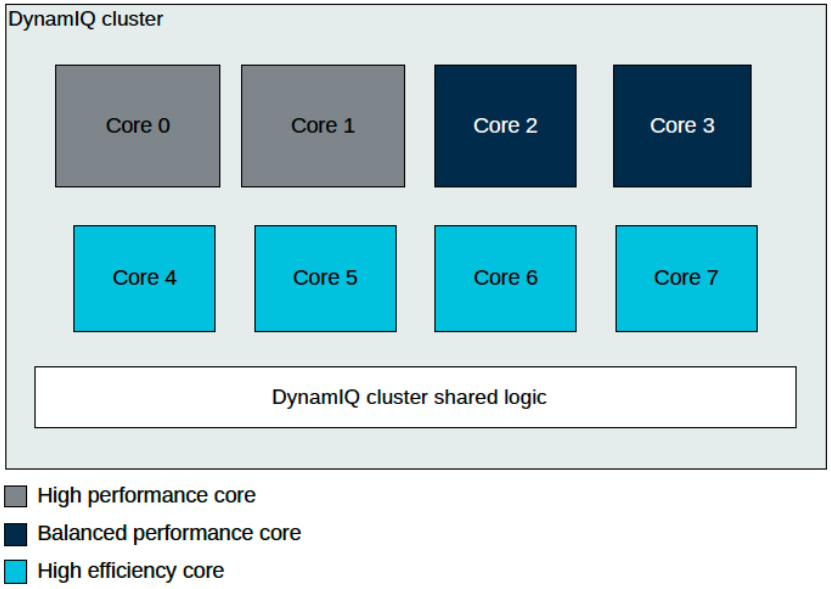
Claim 1	Accused Products

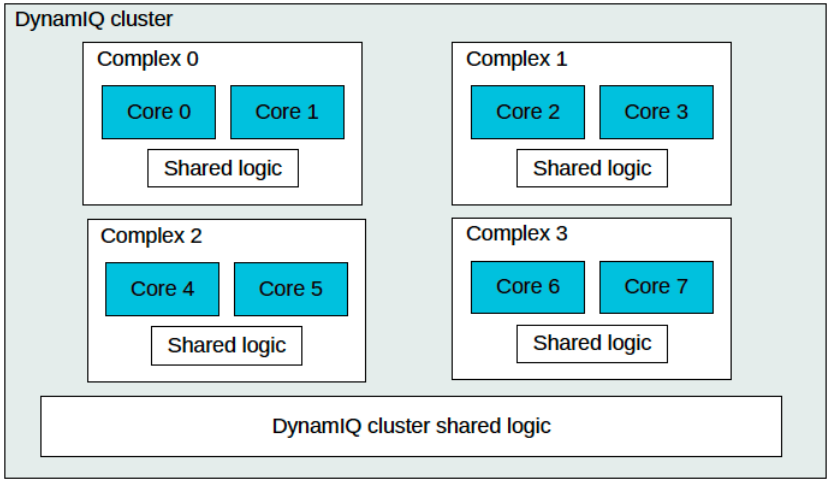
Model number	Product Name	Fab	Die size	CPU
MSM8998[246]	Snapdragon 835	10 nm FinFET (Samsung 10LPE)	72.3 mm^2	4 + 4 cores Kryo 280 (2.45 GHz Cortex-A73 + 1.9 GHz Cortex-A53)

https://en.wikipedia.org/wiki/List_of_Qualcomm_Snapdragon_systems_on_chips#Snapdragon_800_series_(2013%E2%80%932021)

Claim 1	Accused Products
	Cluster features The DSU-110 has the following cluster features: • Support for Arm®v9.0-A architecture cores • Support for up to four types of core, and a maximum of 12 cores in the cluster • *Power Policy Units* (PPUs) providing autonomous power management of the L3 cache and the cores • Support for cores running independently at different frequencies and voltages known as *Dynamic Voltage Frequency Scaling* (DVFS). For cores in a complex, DVFS is only possible for the whole complex not for individual cores. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual, available at https://documentation-service.arm.com/static/62bb28beb334256d9ea8cc32, at p. 19 A cluster can be configured with up to four different types of cores in the same cluster. Each core type targeting different power efficiency and performance levels. This arrangement allows for an intermediate core that has an intermediate performance and efficiency level. The cluster also supports complexes. A cluster can be configured in many arrangements. Examples of cluster arrangements are: • One or more cores of the same type. • Various arrangements of two types of cores. For example, one or more cores targeting either a high-performance level or a higher power efficiency level. • Various arrangements of three or four types of cores. For example, one or more high-peformance cores, power-efficient cores, and intermediate cores. • One or more complexes and no individual cores. For information on complexes, see 2.3.1 What is a complex? on page 25. • One or more complexes and individual cores. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 22

Claim 1	Accused Products
	 Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 23

Claim 1	Accused Products
	 Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 24

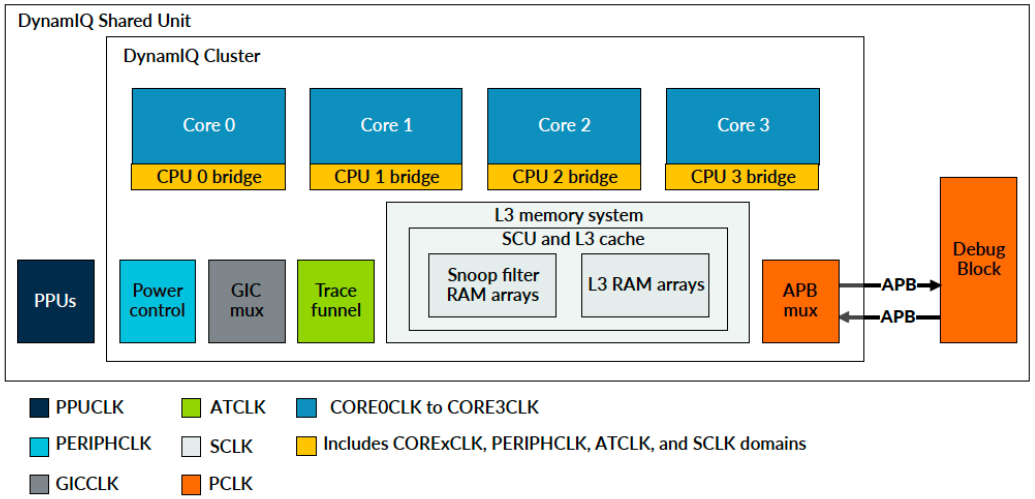
Claim 1	Accused Products
	 Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 24 The DSU-110 DynamIQ™ cluster supports blocks that are called complexes which contain up to two cores of the same type and some shared logic. Sharing some logic between the two cores of a dual core complex can make the dual core complex area efficient. However, this area efficiency is at the cost of reduced performance compared with using two single-core complexes. Arm® DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 25

Claim 1	Accused Products

The *DynamIQ™ Shared Unit-110* (DSU-110) has multiple clock domains. Each core or complex can be implemented in a separate clock domain.

The following figure shows the clock domains for an example cluster with four standalone cores.

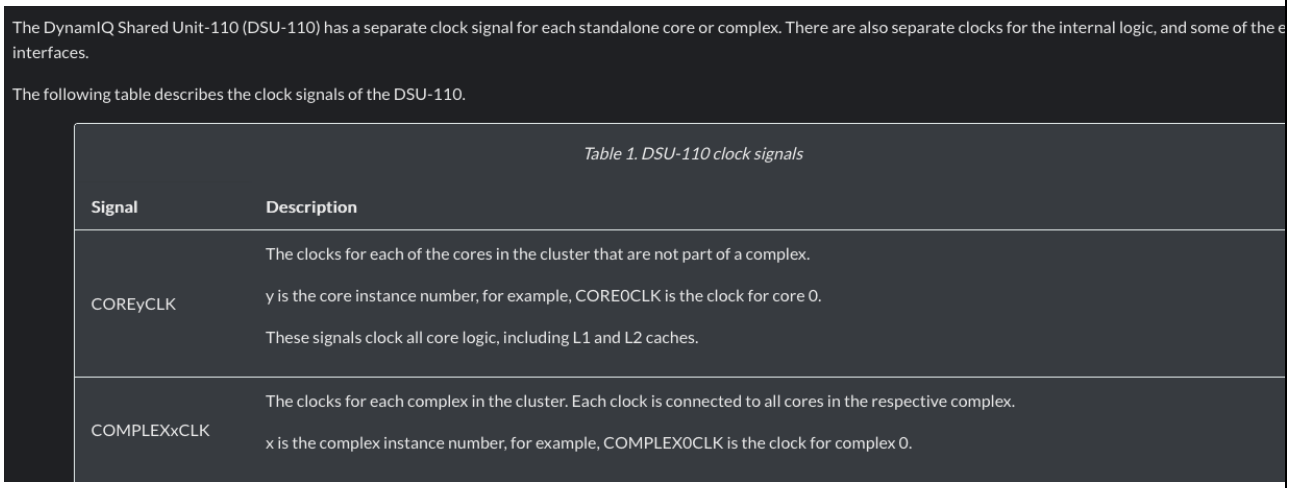
Figure 4-1: DSU-110 clock domains



Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 46

The *DynamIQ™ Shared Unit-110* (DSU-110) *Power Policy Units* (PPUs) control power management for the Cortex®-A510 core. A Cortex®-A510 complex supports separate gated power domains for the complex, for each core inside the complex, and for the *Vector Processing Unit* (VPU). It also supports a dedicated voltage domain for each complex, and a voltage domain for the DSU-110 DynamIQ™ cluster.

Arm Cortex-A510 Core Technical Reference Manual at p. 39

Claim 1	Accused Products
	The cluster contains several clock domains for functionality that is likely to be connected to different clocks in the system. Within each core, the CPU bridge contains asynchronous bridges fo all crossings between the core and cluster clock domains. Each CPU bridge is split, with one half of each bridge in the core clock domain and the other half in the relevant cluster domain. At the cluster level, there is the *Snoop Control Unit* (SCU) bridge which contains crossings between the cluster clock domains as required. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 46  The DynamIQ Shared Unit-110 (DSU-110) has a separate clock signal for each standalone core or complex. There are also separate clocks for the internal logic, and some of the e interfaces. The following table describes the clock signals of the DSU-110. *Table 1. DSU-110 clock signals* **Signal** / **Description** COREyCLK: The clocks for each of the cores in the cluster that are not part of a complex. y is the core instance number, for example, CORE0CLK is the clock for core 0. These signals clock all core logic, including L1 and L2 caches. COMPLEXxCLK: The clocks for each complex in the cluster. Each clock is connected to all cores in the respective complex. x is the complex instance number, for example, COMPLEX0CLK is the clock for complex 0. https://developer.arm.com/documentation/101381/0400/Clocks-and-resets/Clocks-

Claim 1	Accused Products
	4.1.1 Input clocks Input clocks must be provided with a source that is external to, or embedded in, the subsystem. **Table 4-1: Input clocks** | Clock	Description | |---|---| | REFCLK | Main reference clock The input clock to *System Control Processor* (SCP) boots coming out of reset. REFCLK also clocks other logic like SCP generic and watchdog timers. | | CPUxPLLCLKn | CPUxPLLCLKn is a high-frequency reference clock inputs used for application core PLL, that is, one per core, where n = 0-7. | | CLUS0PLLCLKn | CLUS0PLLCLKn is a high-frequency reference clock inputs used for cluster PLL, that is, one clock per cluster, where n is the number of clusters in a subsystem. As RD-TC21 reference subsystem only contain one cluster, n is fixed to 1. | | INTPLLCLK | INTPLLCLK is a high-frequency reference clock input used for the Interconnect block. | | SYSPLLCLK | SYSPLLCLK is a high-frequency clock input for the main block. | | DMCPLLCLK | DMCPLLCLK is a high-frequency reference clock input used for the Memory block. | | GPUPLLCLK | GPUPLLCLK is a high-frequency reference clock input used for the GPU block. | Arm® Total Compute 2021 Reference Design Software Developer Guide at p. 40 4.1.2 Systems clocks RD-TC21 internally derives clocks that are used for parts of the subsystem. These clocks are generated only when the VSYS.SYSTOP power domain is powered. Each clock can be individually controlled. The following table summarizes these derived internal clocks. *Id.*

Claim 1	Accused Products

Table 4-2: Systems internal derived clocks

Clock Signal	PLL	Description
CLUSxCORECLKn n is the number of clusters in a subsystem.	CPUPLL	Core clock. Each core is clocked independently to each other. Therefore, One clock per cluster, where
CLUSxCLK	CLUSxPLLCLKn	CPU Cluster x clock. The cluster clock drives the logic in the Processor block, where x is 0 to 31.
INTCLKOUT	INTPLLCLK	Coherent Interconnect Clock Clock for the interconnect and other expansion master and slave ports. This clock is generated from INTPLLCLK

Id. at p. 41

4.3 Power control

Good power management is key to reduce system power consumption while maintaining a high performance.

An RD-TC21 reference subsystem contains the following power management features:

- Multiple voltage domains to allow for *Dynamic Voltage and Frequency Scaling* (DVFS) on application processors and the GPU
- Multiple power-gated regions provide comprehensive leakage management
- Multiple power modes for different system scenarios
- A *System Control Processor* (SCP) based on a Cortex®-M3 processor controls power, clock, reset, and the static configuration of the system
- *Power Policy Units* (PPUs) are used to manage power states of each voltage and power domain under the control of the SCP

Id. at 47-48

Claim 1	Accused Products
	4.3.1 Voltage domain A voltage domain is defined as a collection of design elements supplied by a single voltage. The voltage supply to the domain might be scaled or switched off for power or performance reasons. Total Compute 2021 Reference Design (RD-TC21) supports the following voltage domains: **VCPU0** The first voltage domain in the processor cluster for "LITTLE" cores. Supports DVFS. **VCPU1** The second voltage domain in the processor cluster for ELP and "big" cores. Supports DVFS. **VGPU** The voltage domain for GPU. Supports DVFS. **VSYS** The voltage domain for the rest of the subsystem. Does not support DVFS. *Id.* at p. 48 **SCP runtime firmware** The SCP runtime firmware executes after *Trusted Firmware for A-profile* (TF-A) BL2 has authenticated and copied it from flash. The SCP runtime firmware performs the following functions: • Responds to *System Control and Management Interface* (SCMI) messages through *Message Handling Unit* (MHU) version 2.0 for processor power control and *Dynamic Voltage and Frequency Scaling* (DVFS) • Power domain management • Clock management *Id.* at p. 65

Claim 1	Accused Products
	Our initial testing shows that Snapdragon 835's Kryo 280 CPU is an octa-core, big.LITTLE configuration with four semi-custom A73 "performance" cores and four semi-custom A53 "efficiency" cores. Kryo 280's performance cores are pretty much equivalent to Kirin 960's A73 cores in both integer and floating-point IPC, but comparing them to Snapdragon 820's Kryo CPU shows mixed results: integer IPC improves but floating-point regresses. https://www.anandtech.com/show/11201/qualcomm-snapdragon-835-performance-preview/6 **Voltage domains** The voltage supply to a domain might be scaled or removed for power or performance reasons. Except for low complexity solutions, it is rare to use a single logic voltage supply for the whole SoC. The primary reason for additional voltage domains is to support DVFS for functional areas of the SoC. The second reason is to enable external supply switch-off, or reduction to non-functional state retention levels, to some logic areas while maintaining an operational level supply to others. However, the cost for additional voltage domains is significant, because additional voltage regulators, extra effort, and complexity are required in the SoC physical implementation. Therefore, you must carefully assess the value of the addition of each voltage domain against the performance and power requirements for the design. In a big.LITTLE system, each cluster must have a dedicated voltage supply. This is a critical success factor when combined with big.LITTLE software. "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products

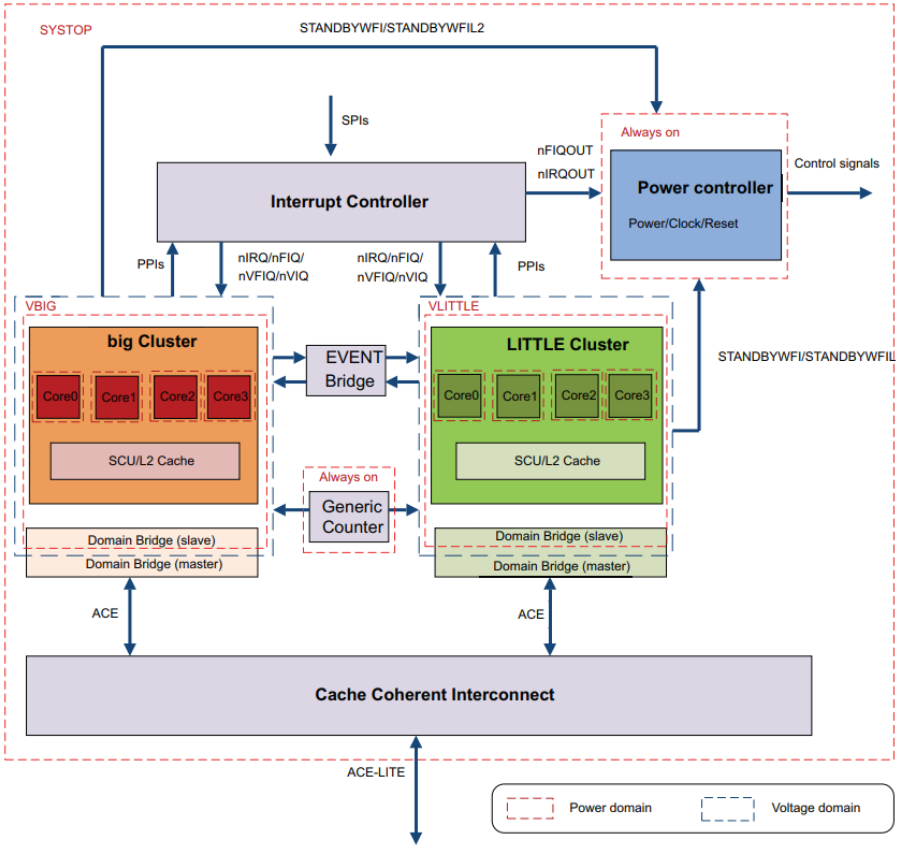
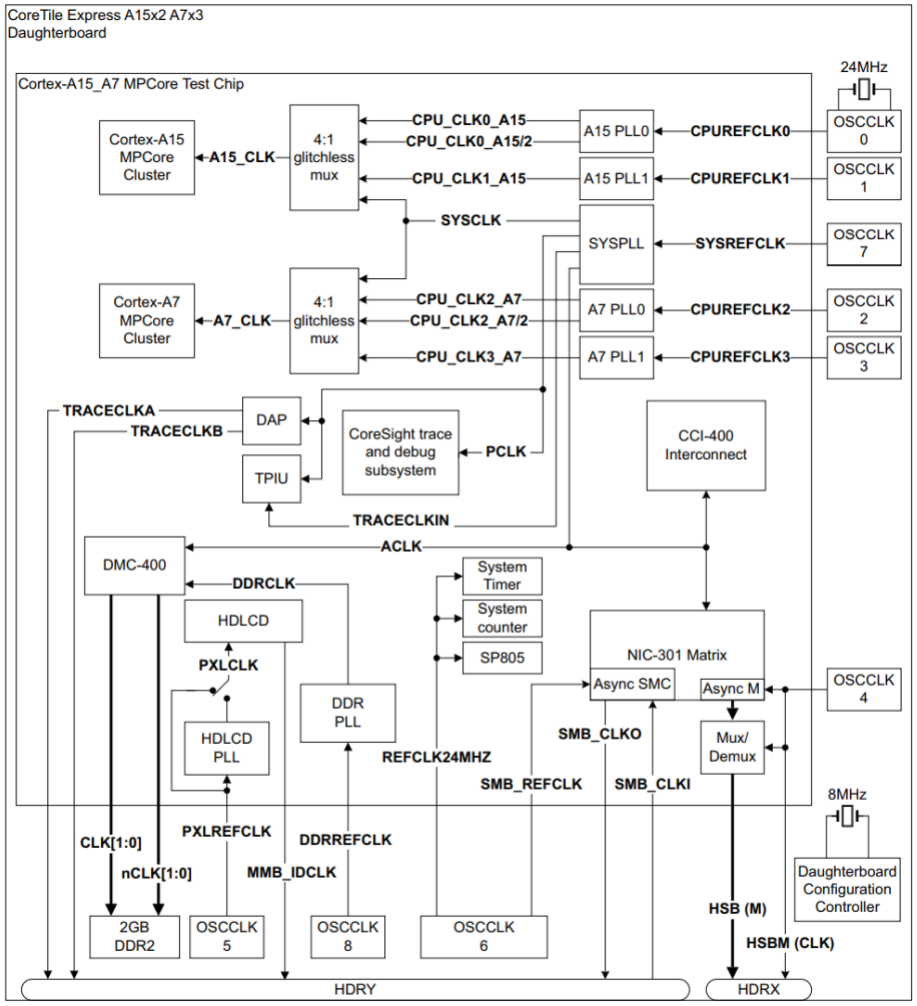


Figure 2-2 Power domain hierarchy

In Figure 2-2, two processor clusters are implemented with per-core and cluster power domains. The big cluster is in the VBIG voltage domain, while the LITTLE cluster is in the VLITTLE voltage domain. The system controller is in an always-on power domain. All the other components are in the system logic power domain (SYSTOP).

"High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
	Clock domains Clock domains can interact with each other synchronously or asynchronously. Synchronous clock domains can have independent source activity. Each cluster requires an independent clock, and the CCI requires a clock. "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
	 **Figure 2-10 CoreTile Express A15×2 A7×3 daughterboard clocks** ARM® CoreTile Express A15×2 A7×3 Technical Reference Manual, available at https://developer.arm.com/documentation/ddi0503/i/

Claim 1	Accused Products
[1b] a second set of processor cores of the multi-core processor, wherein each processor core from the second set of processor cores is configured to dynamically receive a second supply voltage and a second output clock signal of a second PLL having a second clock signal as input, wherein the first supply voltage is independent from the second supply voltage, and the first clock signal is independent from the second clock signal; and	Each Accused Product comprises a second set of processor cores of the multi-core processor, wherein each processor core from the second set of processor cores is configured to dynamically receive a second supply voltage and a second output clock signal of a second PLL having a second clock signal as input, wherein the first supply voltage is independent from the second supply voltage, and the first clock signal is independent from the second clock signal. For example, in addition to the first set of processor cores, the Snapdragon 8 Gen 2 Octa-Core has a second set of high-performance processor cores (two or more ARM Cortex-A710 cores). For another example, in addition to the first set of processor cores, the Snapdragon 8 Gen 2 Octa-Core has a second set of high-performance processor cores (two or more ARM Cortex-A715 cores). The processor cores in the second set receive a second, independent dynamic supply voltage and a second clock signal of a second PLL receiving an independent second clock signal. For a further example, in addition to the first set of processor cores, the Snapdragon 835 Mobile Platform has a second set of high-performance processor cores (two or more A73 cores). The processor cores in the second set receive a second, independent dynamic supply voltage and a second clock signal of a second PLL receiving an independent second clock signal. ARM documentation for the big.LITTLE and DynamIQ architectures used in the Accused Products directly shows that each core cluster receives its own clock domain. At the time the Accused Products were designed, it was typical to produce this clock using a PLL that has a corresponding clock input. Furthermore, ARM documentation for an earlier, related device (the Cortex-A15_A7 MPCore test chip, which also has independent clock domains for different CPU clusters) shows each CPU cluster receiving an output clock signal from a PLL having a corresponding clock signal as input (*e.g.* from an oscillator). It is therefore substantially likely that each Accused Product specifically receives a second output clock signal of a second PLL having a second clock signal as input. *See, e.g.*:

Claim 1	Accused Products

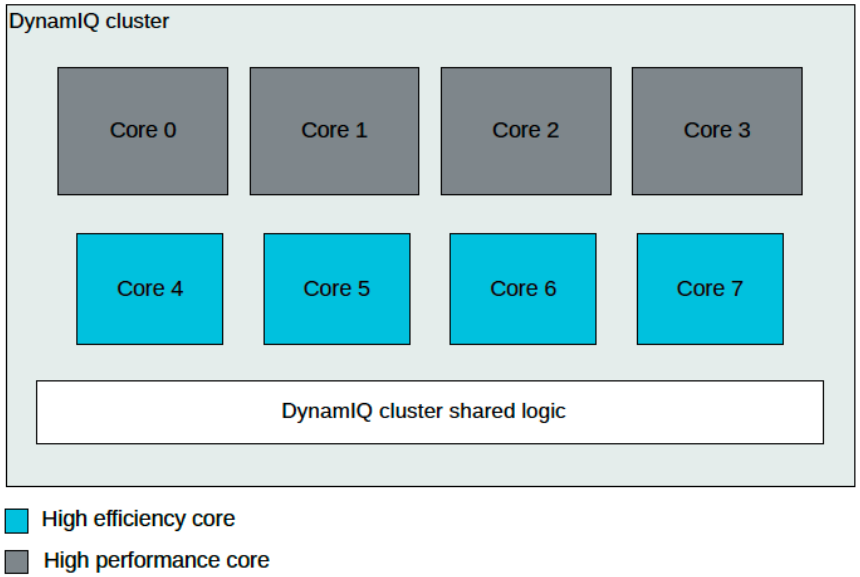
	Snapdragon 8 Gen 2 (2023) [edit]
	The **Snapdragon 8 Gen 2** was announced on November 15, 2022.[252]
	Notable features over its predecessor (8 Gen 1):
	• 4nm (TSMC N4) process
	• Support up to 16 GB LPDDR5X 4200 MHz
	• Support UFS 4.0
	• CPU features
	  • 1 Kryo Prime (ARM Cortex-X3), up to 3.36 GHz. Prime core
	  • 1MB L2 cache
	  • Only 64-bit support [253]
	  • 2 Kryo Gold (ARM Cortex-A715), up to 2.8 GHz. High Performance cores
	  • Only 64-bit support [253]
	  • 2 Kryo Gold (ARM Cortex-A710), up to 2.8 GHz. Performance cores
	  • 32-bit and 64-bit support[253]
	  • 3 Kryo Silver (ARM Cortex-A510), up to 2 GHz. Efficiency cores
	  • 32-bit and 64-bit support[253]
	  • 35% performance uplift and 40% power efficiency improvement
	  • 8 MB system-level cache
	https://en.wikipedia.org/wiki/List_of_Qualcomm_Snapdragon_systems_on_chips#Snapdragon_8_Gen_2_(2023)

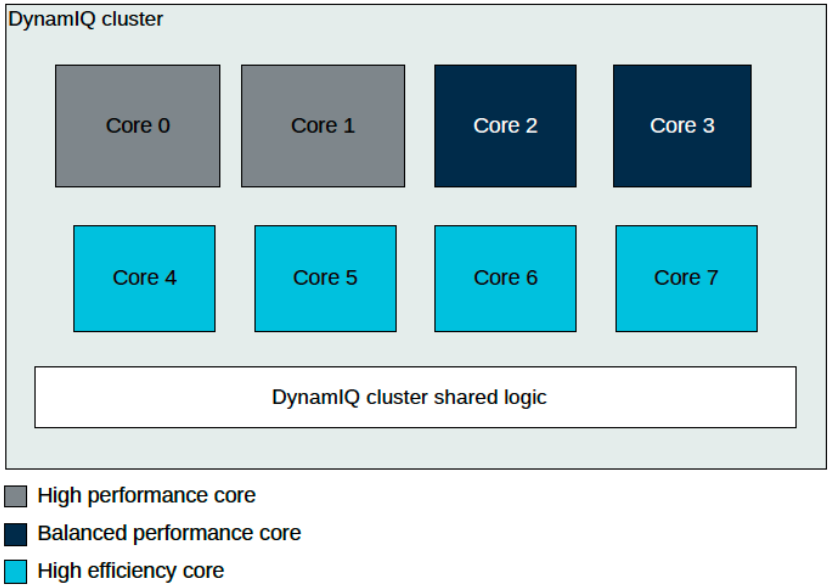
Claim 1	Accused Products

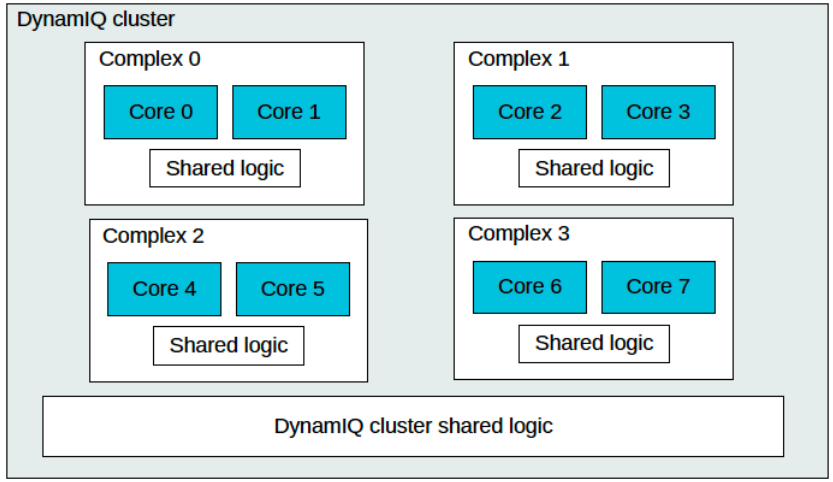
Model number	Product Name	Fab	Die size	CPU
MSM8998[246]	Snapdragon 835	10 nm FinFET (Samsung 10LPE)	72.3 mm^2	4 + 4 cores Kryo 280 (2.45 GHz Cortex-A73 + 1.9 GHz Cortex-A53)

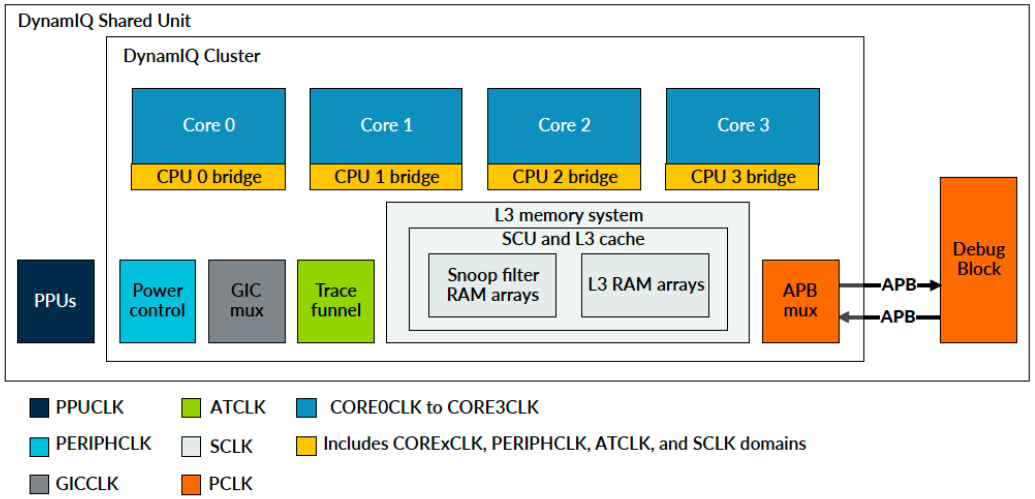
https://en.wikipedia.org/wiki/List_of_Qualcomm_Snapdragon_systems_on_chips#Snapdragon_800_series_(2013%E2%80%932021)

Claim 1	Accused Products
	Cluster features The DSU-110 has the following cluster features: • Support for Arm®v9.0-A architecture cores • Support for up to four types of core, and a maximum of 12 cores in the cluster • *Power Policy Units* (PPUs) providing autonomous power management of the L3 cache and the cores • Support for cores running independently at different frequencies and voltages known as *Dynamic Voltage Frequency Scaling* (DVFS). For cores in a complex, DVFS is only possible for the whole complex not for individual cores. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual, available at https://documentation-service.arm.com/static/62bb28beb334256d9ea8cc32, at p. 19 A cluster can be configured with up to four different types of cores in the same cluster. Each core type targeting different power efficiency and performance levels. This arrangement allows for an intermediate core that has an intermediate performance and efficiency level. The cluster also supports complexes. A cluster can be configured in many arrangements. Examples of cluster arrangements are: • One or more cores of the same type. • Various arrangements of two types of cores. For example, one or more cores targeting either a high-performance level or a higher power efficiency level. • Various arrangements of three or four types of cores. For example, one or more high-peformance cores, power-efficient cores, and intermediate cores. • One or more complexes and no individual cores. For information on complexes, see 2.3.1 What is a complex? on page 25. • One or more complexes and individual cores. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 22

Claim 1	Accused Products
	 Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 23

Claim 1	Accused Products
	 Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 24

Claim 1	Accused Products
	 Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 24 The DSU-110 DynamIQ™ cluster supports blocks that are called complexes which contain up to two cores of the same type and some shared logic. Sharing some logic between the two cores of a dual core complex can make the dual core complex area efficient. However, this area efficiency is at the cost of reduced performance compared with using two single-core complexes. Arm® DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 25

Claim 1	Accused Products
	The *DynamIQ™ Shared Unit-110* (DSU-110) has multiple clock domains. Each core or complex can be implemented in a separate clock domain. The following figure shows the clock domains for an example cluster with four standalone cores. **Figure 4-1: DSU-110 clock domains**  Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 46

Claim 1	Accused Products
	6.1 The Power Policy Unit Power mode control for the *DynamIQ™ Shared Unit-110* (DSU-110) is provided by the *Power Policy Units* (PPUs) that are integrated into the cluster. These PPUs control all the PPU modes for all components in the cluster. A PPU is a standard component for abstracting software-controlled power domain policy to low-level hardware control signaling. There is one PPU for controlling the DSU-110 DynamIQ™ cluster power domain (PDCLUSTER). Also, each core has its own individual PPU for controlling its respective core power domain (for example, a PPU for PDCORE0 and a PPU for PDCORE1). This includes any cores included as part of a complex. A component in the system such as a *System Control Processor* (SCP) can program the PPUs through the utility bus to set the required power policy. The PPUs control the low-level details of powering up, powering down, and resetting domains as necessary to implement the requested policy. The hardware performs any actions to reach the requested power mode, such as gating clocks, cleaning and invalidating caches, or disabling coherency. Note • Although the cluster and each core in the cluster has their own PPU, the shared logic of a complex does not have a dedicated PPU. Instead, power management of the complex is controlled as a combination of the PPUs for the cores it contains. See Table 5-8: PPU mode and power domain states for a dual-core complex on page 74. • The cluster and all the core PPUs are provided as part of the DSU-110. • The implementation process automatically creates the PPU for the cluster and each core PPU, and connects these into the DSU-110 DynamIQ™ cluster. Each PPU has a set of memory-mapped control registers which is accessed using the utility bus. *Id.* at p. 78

Claim 1	Accused Products

The *DynamIQ™ Shared Unit-110* (DSU-110) *Power Policy Units* (PPUs) control power management for the Cortex®-A510 core. A Cortex®-A510 complex supports separate gated power domains for the complex, for each core inside the complex, and for the *Vector Processing Unit* (VPU). It also supports a dedicated voltage domain for each complex, and a voltage domain for the DSU-110 DynamIQ™ cluster.

Arm Cortex-A510 Core Technical Reference Manual at p. 39

The cluster contains several clock domains for functionality that is likely to be connected to different clocks in the system. Within each core, the CPU bridge contains asynchronous bridges fo all crossings between the core and cluster clock domains. Each CPU bridge is split, with one half of each bridge in the core clock domain and the other half in the relevant cluster domain. At the cluster level, there is the *Snoop Control Unit* (SCU) bridge which contains crossings between the cluster clock domains as required.

Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 46

The DynamIQ Shared Unit-110 (DSU-110) has a separate clock signal for each standalone core or complex. There are also separate clocks for the internal logic, and some of the e interfaces.

The following table describes the clock signals of the DSU-110.

Table 1. DSU-110 clock signals

Signal	Description
COREyCLK	The clocks for each of the cores in the cluster that are not part of a complex. y is the core instance number, for example, CORE0CLK is the clock for core 0. These signals clock all core logic, including L1 and L2 caches.
COMPLEXxCLK	The clocks for each complex in the cluster. Each clock is connected to all cores in the respective complex. x is the complex instance number, for example, COMPLEX0CLK is the clock for complex 0.

https://developer.arm.com/documentation/101381/0400/Clocks-and-resets/Clocks-

Claim 1	Accused Products
	4.1.1 Input clocks Input clocks must be provided with a source that is external to, or embedded in, the subsystem. **Table 4-1: Input clocks** | Clock	Description | |---|---| | REFCLK | Main reference clock The input clock to *System Control Processor* (SCP) boots coming out of reset. REFCLK also clocks other logic like SCP generic and watchdog timers. | | CPUxPLLCLKn | CPUxPLLCLKn is a high-frequency reference clock inputs used for application core PLL, that is, one per core, where n = 0-7. | | CLUS0PLLCLKn | CLUS0PLLCLKn is a high-frequency reference clock inputs used for cluster PLL, that is, one clock per cluster, where n is the number of clusters in a subsystem. As RD-TC21 reference subsystem only contain one cluster, n is fixed to 1. | | INTPLLCLK | INTPLLCLK is a high-frequency reference clock input used for the Interconnect block. | | SYSPLLCLK | SYSPLLCLK is a high-frequency clock input for the main block. | | DMCPLLCLK | DMCPLLCLK is a high-frequency reference clock input used for the Memory block. | | GPUPLLCLK | GPUPLLCLK is a high-frequency reference clock input used for the GPU block. | Arm® Total Compute 2021 Reference Design Software Developer Guide at p. 40 4.1.2 Systems clocks RD-TC21 internally derives clocks that are used for parts of the subsystem. These clocks are generated only when the VSYS.SYSTOP power domain is powered. Each clock can be individually controlled. The following table summarizes these derived internal clocks. *Id.*

Claim 1	Accused Products

Table 4-2: Systems internal derived clocks

Clock Signal	PLL	Description
CLUSxCORECLKn n is the number of clusters in a subsystem.	CPUPLL	Core clock. Each core is clocked independently to each other. Therefore, One clock per cluster, where
CLUSxCLK	CLUSxPLLCLKn	CPU Cluster x clock. The cluster clock drives the logic in the Processor block, where x is 0 to 31.
INTCLKOUT	INTPLLCLK	Coherent Interconnect Clock Clock for the interconnect and other expansion master and slave ports. This clock is generated from INTPLLCLK

Id. at p. 41

4.3 Power control

Good power management is key to reduce system power consumption while maintaining a high performance.

An RD-TC21 reference subsystem contains the following power management features:

- Multiple voltage domains to allow for *Dynamic Voltage and Frequency Scaling* (DVFS) on application processors and the GPU
- Multiple power-gated regions provide comprehensive leakage management
- Multiple power modes for different system scenarios
- A *System Control Processor* (SCP) based on a Cortex®-M3 processor controls power, clock, reset, and the static configuration of the system
- *Power Policy Units* (PPUs) are used to manage power states of each voltage and power domain under the control of the SCP

Id. at 47-48

Claim 1	Accused Products
	4.3.1 Voltage domain A voltage domain is defined as a collection of design elements supplied by a single voltage. The voltage supply to the domain might be scaled or switched off for power or performance reasons. Total Compute 2021 Reference Design (RD-TC21) supports the following voltage domains: **VCPU0** The first voltage domain in the processor cluster for "LITTLE" cores. Supports DVFS. **VCPU1** The second voltage domain in the processor cluster for ELP and "big" cores. Supports DVFS. **VGPU** The voltage domain for GPU. Supports DVFS. **VSYS** The voltage domain for the rest of the subsystem. Does not support DVFS. *Id.* at p. 48 **SCP runtime firmware** The SCP runtime firmware executes after *Trusted Firmware for A-profile* (TF-A) BL2 has authenticated and copied it from flash. The SCP runtime firmware performs the following functions: • Responds to *System Control and Management Interface* (SCMI) messages through *Message Handling Unit* (MHU) version 2.0 for processor power control and *Dynamic Voltage and Frequency Scaling* (DVFS) • Power domain management • Clock management *Id.* at p. 65

Claim 1	Accused Products
	Our initial testing shows that Snapdragon 835's Kryo 280 CPU is an octa-core, big.LITTLE configuration with four semi-custom A73 "performance" cores and four semi-custom A53 "efficiency" cores. Kryo 280's performance cores are pretty much equivalent to Kirin 960's A73 cores in both integer and floating-point IPC, but comparing them to Snapdragon 820's Kryo CPU shows mixed results: integer IPC improves but floating-point regresses. https://www.anandtech.com/show/11201/qualcomm-snapdragon-835-performance-preview/6 **Voltage domains** The voltage supply to a domain might be scaled or removed for power or performance reasons. Except for low complexity solutions, it is rare to use a single logic voltage supply for the whole SoC. The primary reason for additional voltage domains is to support DVFS for functional areas of the SoC. The second reason is to enable external supply switch-off, or reduction to non-functional state retention levels, to some logic areas while maintaining an operational level supply to others. However, the cost for additional voltage domains is significant, because additional voltage regulators, extra effort, and complexity are required in the SoC physical implementation. Therefore, you must carefully assess the value of the addition of each voltage domain against the performance and power requirements for the design. In a big.LITTLE system, each cluster must have a dedicated voltage supply. This is a critical success factor when combined with big.LITTLE software. "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products

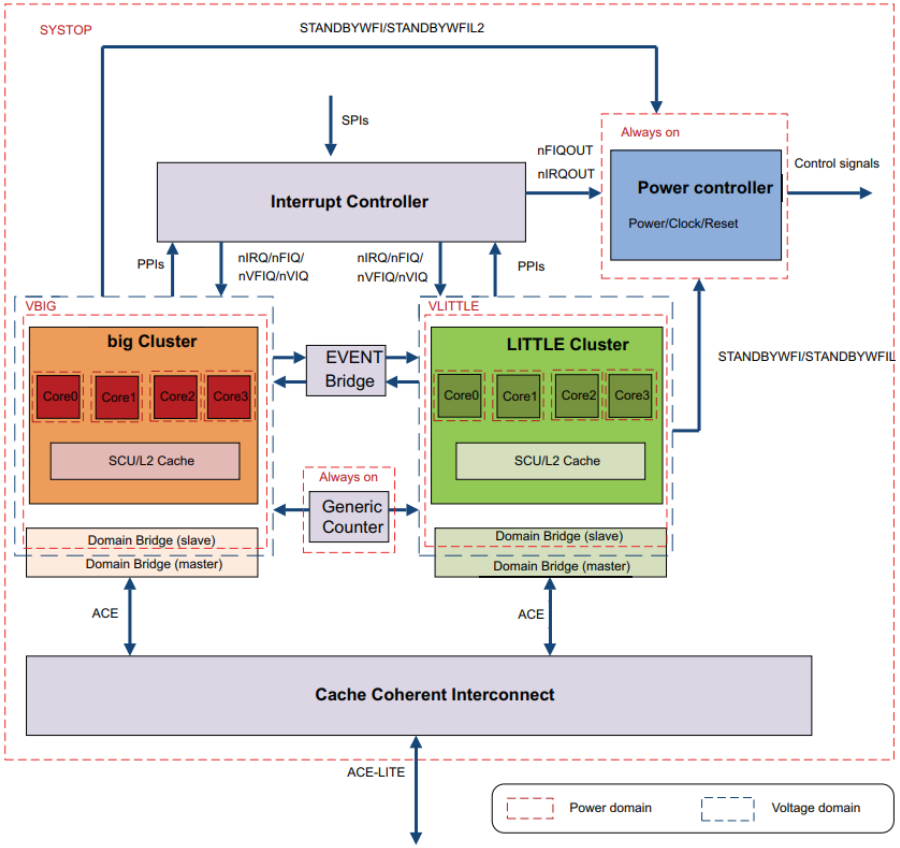
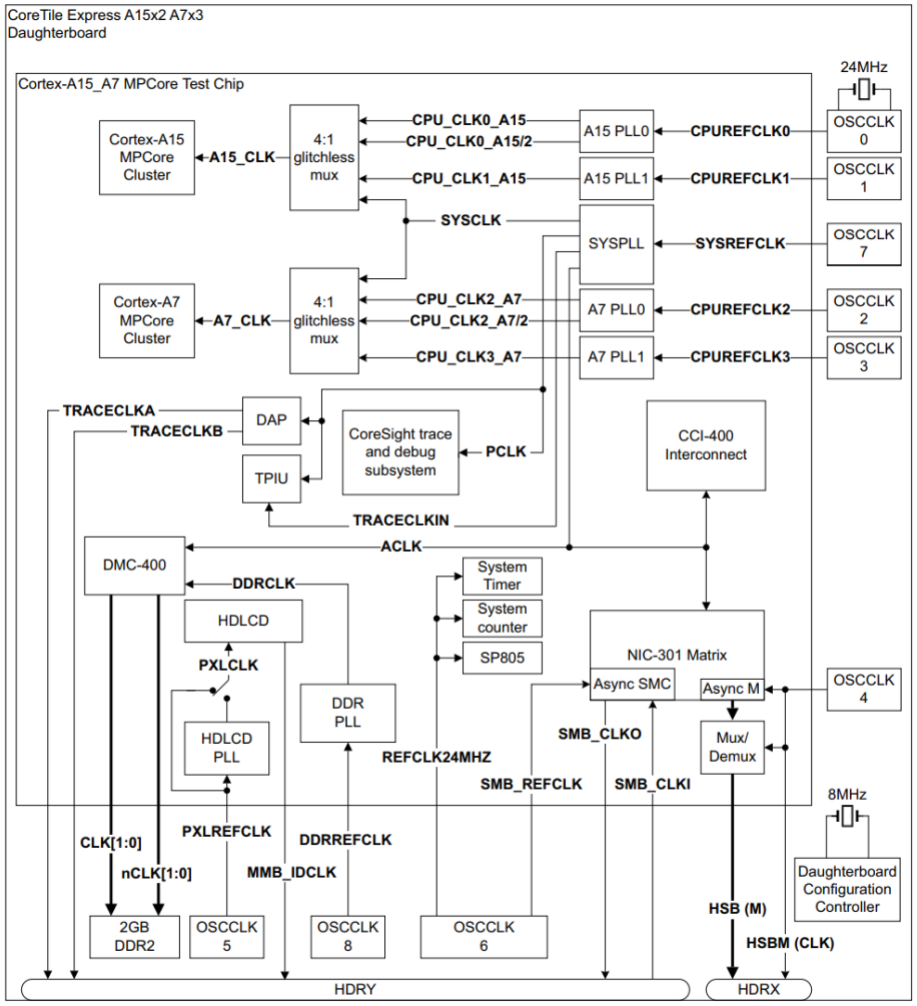


Figure 2-2 Power domain hierarchy

In Figure 2-2, two processor clusters are implemented with per-core and cluster power domains. The big cluster is in the VBIG voltage domain, while the LITTLE cluster is in the VLITTLE voltage domain. The system controller is in an always-on power domain. All the other components are in the system logic power domain (SYSTOP).

"High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
	Clock domains Clock domains can interact with each other synchronously or asynchronously. Synchronous clock domains can have independent source activity. Each cluster requires an independent clock, and the CCI requires a clock. "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

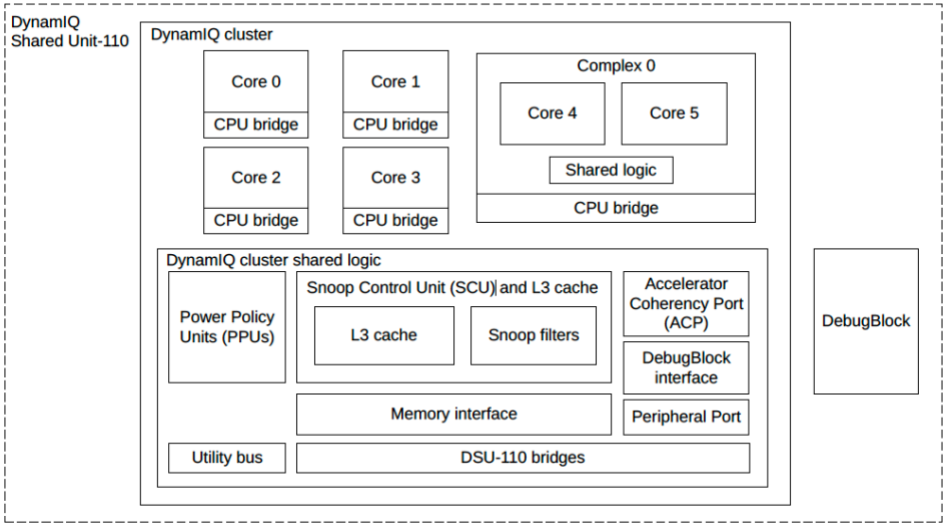
Claim 1	Accused Products
	 **Figure 2-10 CoreTile Express A15×2 A7×3 daughterboard clocks** ARM® CoreTile Express A15×2 A7×3 Technical Reference Manual, available at https://developer.arm.com/documentation/ddi0503/i/

Claim 1	Accused Products
[1c] an interface block coupled to the first set of processor cores and also coupled to the second set of processor cores, wherein the interface block is configured to facilitate communication between the first set of processor cores and the second set of processor cores.	Each Accused Product comprises an interface block coupled to the first set of processor cores and also coupled to the second set of processor cores, wherein the interface block is configured to facilitate communication between the first set of processor cores and the second set of processor cores. For example, the Snapdragon 8 Gen 2 Octa-Core includes a DynamIQ Shared Unit (DSU), including a Snoop Control Unit (SCU), that is coupled to each core or to each set of processor cores and is configured to communicate between the sets of processor cores, including by buffering and resynchronizing signals across clock domains and by maintaining coherency between caches in the cores or complexes and/or with the L3 shared memory implemented by the DSU. For another example, the Snapdragon 835 Mobile Platform includes a cache coherent interconnect, such as a CCI-400 or CCI-550, that is coupled to each core or to each set of processor cores and is configured to communicate between the sets of processor cores, for example by maintaining coherency between caches in the cores or complexes. As shown in the relevant documentation, described above and reiterated here for the avoidance of all possible doubt, the Accused Products implement either the big.LITTLE or DynamIQ architecture, which are the heterogeneous processing architectures for the ARM Cortex-A53/A72 and ARM Cortex-A510/A710/A715 respectively. The big.LITTLE architecture is also the heterogenous processing architecture for the ARM Cortex-A7 and Cortex-A15 processors. For example, ARM documentation for the big.LITTLE architecture expressly includes all four of these processors in its stated scope. ARM's public documentation of the big.LITTLE architecture describes the infringing cache coherent interconnect in general, and also in the context of a "typical big.LITTLE system" featuring Cortex-A7 and Cortex-A15 processors. There is a reasonable inference that ARM's document also applies to the Accused Products with big.LITTLE architectures including Cortex-A53 and Cortex-A72 cores. And the ARM documentation for the ARM Cortex-A510 processor core expressly states that it includes and supports the DynamIQ Shared Unit (DSU), creating a reasonable inference that

Claim 1	Accused Products
	ARM's DSU documentation also applies to the Accused Products with DynamIQ architectures including Cortex-A510, Cortex-A710, and/or Cortex-A715 cores. *See, e.g.*: The *DynamIQ™ Shared Unit-110* (DSU-110) *Power Policy Units* (PPUs) control power management for the Cortex®-A510 core. A Cortex®-A510 complex supports separate gated power domains for the complex, for each core inside the complex, and for the *Vector Processing Unit* (VPU). It also supports a dedicated voltage domain for each complex, and a voltage domain for the DSU-110 DynamIQ™ cluster. Arm Cortex-A510 Core Technical Reference Manual at p. 39 When you implement a DSU-110 DynamIQ™ cluster, all interfacing between the cores, complexes, and the *DynamIQ™ Shared Unit-110* (DSU-110) is implemented automatically. All the external signal inputs and outputs pass through the DSU-110. The DSU-110 buffers and resynchronizes many of these signals to allow cores and complexes to be clocked at different speeds. The memory interfacing of each core is internally connected to the DSU-110 L3 memory system. Where necessary, the DSU-110 implements additional buffering to compensate for different clock rates of the core and DSU-110 L3 memory system. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 34 All cores in the DSU-110 DynamIQ™ cluster, including those in complexes, are coherently connected to an L3 memory system that includes an L3 cache and a *Snoop Control Unit* (SCU). The SCU maintains coherency between caches in the cores and the L3 cache, and includes a snoop filter to optimize coherency maintenance operations. The shared L3 cache simplifies process migration between the cores. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 18

Claim 1	Accused Products
	Snoop Control Unit The *Snoop Control Unit* (SCU) maintains coherency between all the data caches in the cluster. The SCU contains buffers that can handle direct cache-to-cache transfers between cores without having to read or write data to the L3 cache. Cache line migration enables dirty lines to be moved between cores. Also, there is no requirement to write back transferred cache line data to the L3 cache. The SCU contains a set of snoop filters that track the addresses for locations cached in the core caches. Including the snoop filters means that the SCU does not need to request a look up in the core caches when it receives a coherent memory request. These snoop filters are accessed by coherent requests from the other cores or from the system. If there is a simultaneous hit in the L3 tags and the SCU snoop filters, then the L3 cache normally provides the data in preference to a core. The size of the snoop filter is automatically determined from the configured number of cores and the cache sizes in those cores. Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 35

Claim 1	Accused Products



Arm DynamIQ™ Shared Unit-110 Technical Reference Manual at p. 17

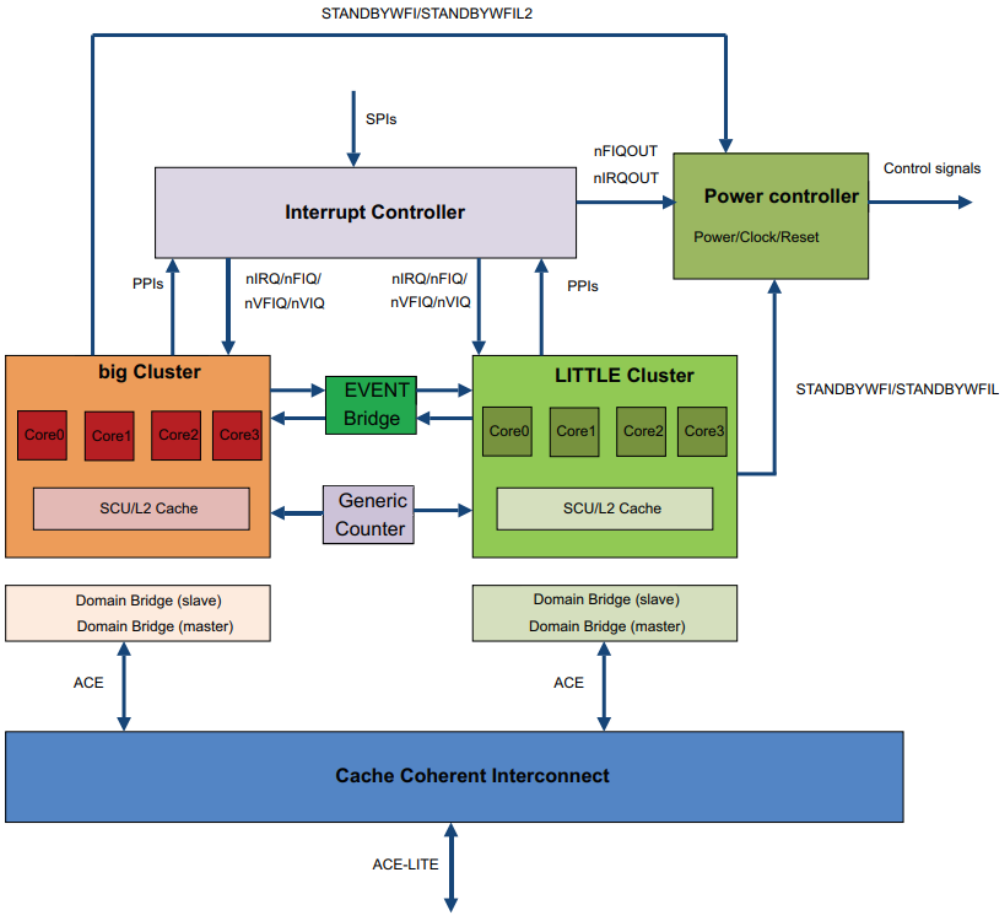
1.2 Document scope

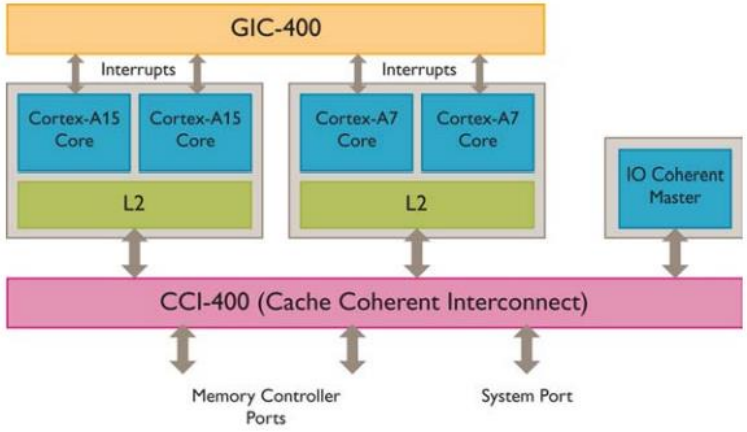
This application note focuses on the following processors and highlights important issues when powering up or powering down processor cores and clusters on an SoC.

- Cortex®-A7.
- Cortex®-A15.
- Cortex®-A17.
- Cortex®-A53.
- Cortex®-A57.
- Cortex®-A72.
- Cortex®-A73.

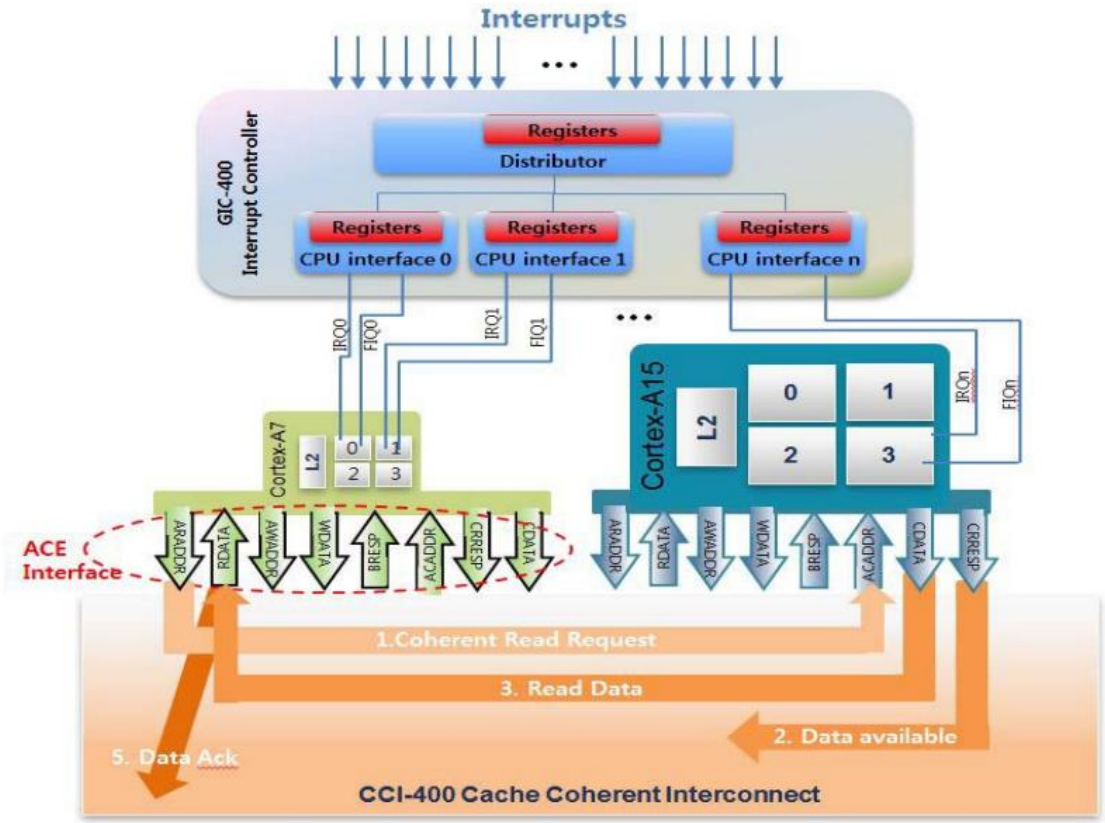
"High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
	2.1 ARM® big.LITTLE™ system example A big.LITTLE system uses two different types of cores that are combined in a coherent system. big cores are designed for high performance while LITTLE cores are designed for high energy efficiency. The big cores are used for resource-intensive software threads, and energy-efficient LITTLE cores handle low-intensity software threads that use fewer compute resources. The result is high energy efficiency and high performance. "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
	 **Figure 2-1 A typical big.LITTLE system** "High-level Considerations for Power Management of a big.LITTLE™ System Application Note 424," available at https://developer.arm.com/documentation/dai0424/a/

Claim 1	Accused Products
	 **Fig 1: Typical big.LITTLE system** As an example, Figure 2 describes the pipeline designs for the Cortex-A15 and Cortex-A7 cores. The Cortex-A15 core is designed to achieve high performance by running more instructions in parallel on a bigger and more complex pipeline. On the other hand, the Cortex-A7 core's pipeline is relatively simple and is designed to be extremely power efficient. The Cortex-A7 core's performance is lower than the Cortex-A15 core's but it is sufficient for most common usage scenarios executed by modern mobile devices. In fact, the Cortex-A7 core's performance is close to Cortex-A9 core, which powers most smartphones today. ARM White Paper "big.LITTLE Technology: The Future of Mobile Making very high performance available in a mobile envelope without sacrificing energy efficiency," available at https://armkeil.blob.core.windows.net/developer/Files/pdf/white-paper/big-little-technology-the-future-of-mobile.pdf

Claim 1	Accused Products
	Cache Coherency Interface and big.LITTLE Technology The key ingredient that makes big.LITTLE technology possible is coherency. big.LITTLE software models require transparent and performant transfer of data between big and LITTLE processors. Hardware coherency enables this, transparently to the software. Without hardware coherency, the transfer of data between big and LITTLE cores would always occur through main memory - this would be slow and not power efficient. In addition, it would require complex cache management software, to enable data conherency between big and LITTLE processors Figure 4 is an example of CPU subsystem consisting of a Cortex-A7 cluster, a Cortex-A15 cluster and a set of system fabric components which enable the seamless data transfer between clusters. This fabric is collectively referred to as a "Cache Coherent Interconnect" – in this case the ARM CoreLink™ CCI-400 interconnect IP. The system is completed by the CoreLink GIC-400, which provides dynamically configurable interrupt distribution to all the cores. ARM White Paper "big.LITTLE Technology: The Future of Mobile Making very high performance available in a mobile envelope without sacrificing energy efficiency," available at https://armkeil.blob.core.windows.net/developer/Files/pdf/white-paper/big-little-technology-the-future-of-mobile.pdf

Claim 1	Accused Products
	 **Figure 3: Cache coherency in a big.LITTLE system** ARM White Paper "big.LITTLE Technology: The Future of Mobile Making very high performance available in a mobile envelope without sacrificing energy efficiency," available at https://armkeil.blob.core.windows.net/developer/Files/pdf/white-paper/big-little-technology-the-future-of-mobile.pdf

Claim 1	Accused Products
	As shown in Figure 3, the bus interfaces of Cortex-A15 and Cortex-A7 processors make use of the AMBA® AXI Coherency Extensions (ACE) to the widely-used AMBA AXI protocol. This protocol provides for coherent data transfer at the bus level. In the AMBA ACE protocol, three coherency channels are added in addition to the normal five channels of AMBA AXI. As an example, the lower part of Figure shows the steps in a coherent data read from the Cortex-A7 cluster to the Cortex-A15 cluster. This starts with the Cortex-A7 cluster issuing a Coherent Read Request through the RADDR channel. The CCI-400 hands over the request to the Cortex-A15 processor's ACADDR channel to snoop into Cortex-A15 processor's cache. On receiving the request from CCI-400, the Cortex-A15 processor checks the data availability and reports this information back through the CRRESP channel. If the requested data is in the cache, the Cortex-A15 processor places it on the CDATA channel. Then the CCI-400 moves the data from the Cortex-A15 processor's CDATA channel to the Cortex-A7 processor's RDATA channel, resulting in a cache linefill in the Cortex-A7 processor. The CCI-400 and the ACE protocol enable full coherency between the Cortex-A15 and Cortex-A7 clusters, allowing data sharing to take place without external memory transactions. ARM White Paper "big.LITTLE Technology: The Future of Mobile Making very high performance available in a mobile envelope without sacrificing energy efficiency," available at https://armkeil.blob.core.windows.net/developer/Files/pdf/white-paper/big-little-technology-the-future-of-mobile.pdf